QUALITY

ASSURANCE

PLAN

Real Estate Assessment Center (REAC)
Financial Assessment Subsystem (FASS-PH)

U.S. Department of Housing and Urban Development

July 11, 2005

System:	REACS
Subsystem:	FASS-PH
Release:	8.1.0.0
Database Release:	REACS
Doc Type:	Quality Assurance Plan
Doc Date:	07/11/05
Doc Author:	Avineon/Virginia Davis
Doc Number:	1.2
Doc Status:	Final

Revision Sheet

Release No.	Date	Revision Description
Rev. 0	5/31/05	Draft QA Plan created by Avineon and submitted for review
Rev. 1	05/31/05	Draft QA Plan revised by Avineon QA Analyst
Rev. 2	06/06/05	Draft QA Plan revised per HUD GTM request
Rev. 3	07/11/05	Updated Final QA Plan revised per project changes

Quality Assurance Plan Authorization Memorandum

I have carefully assessed the Quality Assurance Plan for the <u>FASS-PH subsystem.</u> This document has been completed in accordance with the requirements of the HUD System Development Methodology.

MANAGEMENT CERTIFICATION - Please check the appropriate statement.

_____ The document is accepted.

_____ The document is accepted pending the changes noted.

_____ The document is not accepted.

We fully accept the changes as needed improvements and authorize initiation of work to proceed. Based on our authority and judgment, the continued operation of this system is authorized.

_____   _____
Freddie Harrison   DATE
FASS-PH IT Project Manager

_____   _____
Nicholas Meile   DATE
Business Program Manager

QUALITY ASSURANCE PLAN

TABLE OF CONTENTS

1.0   GENERAL INFORMATION

1   GENERAL INFORMATION

This Quality Assurance Plan (QAP) presents a framework for activities, which when followed, will ensure delivery of quality products and services. The plan provides the project standards and procedures to be used as the basis for the QA group's reviews and audits. This document will be referred to from here forward as the Process and Product Quality Assurance Plan (PPQA).

1.1   Purpose

The purpose of this PPQA Plan is to define the FASS-PH Process and Product Quality Assurance (PPQA) organization, PPQA tasks and responsibilities; implement Avineon's quality policy and objectives providing the FASS-PH subsystem defect-free and delivered on time and within budget, and are maintainable; provide reference documents and guidelines to perform the PPQA activities; provide the standards, practices and conventions used in carrying out PPQA activities; and provide the tools, techniques, and methodologies to support PPQA activities, and PPQA reporting.

The PPQA procedures defined herein shall be used to examine all deliverable software and documentation to determine compliance with technical and performance requirements.

1.2   Scope

This plan establishes the PPQA activities performed throughout the life cycle for the FASS-PH project.

This PPQA plan is written to follow the Avineon PPQA and HUD QA (Quality Assurance) policy and guidelines, for FASS-PH. Specifically, this PPQA Plan will show that the PPQA function is in place for this project. It will show that the PPQA group has a reporting channel to senior management that is independent of the project manager, the project's software engineering group, and software related groups that include Configuration Management (CM), System and Software Test, and Logistics.

This PPQA plan will help assure the following:

(1) That FASS-PH subsystem development, evaluation and acceptance standards are developed, documented and followed per Avineon's PPQA and HUD QA policies and guidelines.

(2) Results from software quality reviews and audits will be submitted to HUD management. This provides feedback as to how well the development effort is conforming to HUD development standards.

(3) Test results adhere to acceptance standards.

Table 1 shows the Avineon Integrated Software Life Cycle activities of the Configuration Items (CIs), as identified by the Institute of Electrical and Electronics Engineers (IEEE)/Electronic Industries Association (EIA) Standard 12207 Series, Software Life Cycle Process, to which this PPQA Plan applies.

ISDLCM PHASES	SOFTWARE LIFECYCLE ACTIVITY	ARTIFACTS
Project Initialization	Customer requirements analysis, Proposal Bid-no-bid decision,	Needs Statement, Feasibility Study, Cost/Benefit Analysis, Risk Analysis, System Decision Paper

Project Planning	Project Planning and Oversight	Project Plan Configuration Management Plan, Quality Assurance Plan, System Support and Acquisition Plan, Functional Requirements Document, Data Requirements Document, System Security Plans, Major Application System, General Support System, Internal Audit Plan, Risk Management Plan
Project Execution	System Requirements Analysis, System Design, Software Requirements Analysis, Software Design, Software Implementation and Unit Testing, Unit Integration and Testing, CI Qualification Testing, CI/Hardware Configuration Item (HWCI) Integration and Testing, System Qualification Testing,	System/Subsystem Specifications, Database Specifications, Program Specifications, Validation, Verification, and Testing Plan, Training Plan, Installation and Conversion Plan, Test Plan (Unit and Integration), User's Manual, Operations Manual, Maintenance Manual, Test Results and Evaluation Report
Project Control	Configuration Management, Quality Assurance, Risk Management, Verification, Validation, Project Monitoring and Control, Requirements Review (RR), Design Review (DR), Specification Review (SR), Preliminary Design Review (PDR), Critical Design Review (CDR), Test Readiness Review (TRR), Formal Qualification Review (FQR), Production Readiness Review (PRR).	CM Plan update, QA Plan update, Risk Management Plan Update, Status reports, review reports.
Project Closing	Software Use Preparation, Software Transition Preparation, Life Cycle Maintenance, Contract Closure, Award, Staff Relocation, Lessons Learned, Senior Management Report.	User Manuals, Transition Plan, Maintenance Plan, Project Summary, Staff Relocation Plan, Lessons Learned, Senior Management Report.

1.3   System Overview

The FASS-PH is a subsystem of the Real Estate Assessment Center System (REACS). FASS-PH will help enable centralized financial analysis that can be used to identify where HUD should focus its limited resources to improve service delivery and manage its housing programs proactively. To achieve this goal, the following objectives have been identified:

- Gather standard financial data pertaining to each Public Housing Agency (PHA) and Section 8 Entity by combining standard fiscal audit information with reporting and compliance factors as defined by the Single Audit Act;
- Assess the financial condition of all PHAs and Section 8 Entities using a comprehensive protocol;

- Assess financial risk using standard financial data;
- Determine an objective, numerical score for each PHA and Section 8 Entity using standard protocols for financial performance review;
- Enable HUD staff to focus on the most troubled PHAs and Section 8 Entities based on the risk associated with the score;
- Eliminate or address existing material weaknesses identified through IG Audits. This includes mitigating potential risks;
- Support HUD's mission;
- Implement OMB Circular A-123 compliant policies and procedures;
- Support HUD's eGov Strategic Plan;
- Automate paper based forms to support the Government Paperwork Elimination Act (GPEA);
- Provide payback as early in the system lifecycle as possible;
- Provide significant benefits to HUD;
- All new functionality meets the Rehabilitation Act Section 508 requirements.

1.3.1 System General Environment

The following table identifies the general environment for the development of FASS-PH:

System Environment Table	
Environment	**System / Organization**
System	Real Estate Assessment Center System (REACS)
Subsystem	Financial Assessment Subsystem - Public Housing (FASS-PH)
Responsible Party Description	
Sponsor	Public and Indian Housing – Real Estate Assessment Center (PIH-REAC)
Requirements	Avineon Inc.
Design	Avineon Inc.
Development	Avineon Inc.
System and Integration Testing	Avineon Inc., DCG
User Acceptance Testing	To be determined by PIH-REAC Management
Deployment	Avineon Inc., DCG
Maintenance	Avineon Inc., DCG
System Environment, Code, and Category: and Operational Status Description	
PCAS	307820
System Code	P093
System Category	Non-Major
Operational Status	Operational
System Environment	Web Based

1.4   Project References

PPQA evaluation of the software development processes throughout the life cycle is based on the processes defined in HUD's software development methodology, as well as on Avineon's CMM standard practices. The PPQA Plan is implemented in conjunction with the FASS-PH Configuration Management Plan.

1.4.1   Reference Documents

This section lists the documents referenced in this PPQA Plan.

    a.   HUD Task Order Request (TOR) GSC-TFMG-05-31210

    b.   HUD Quality Assurance Guidelines

    c.   HUD Software Development Methodology

    d.   HUD Quality Assurance Template

    e.   FASS-PH Project Management Plan

    f.   FASS-PH Configuration Management Plan

    g.   FASS-PH Risk Management Plan

    h.   Avineon Quality Assurance Policy

    i.   Avineon Customer QA Plan Template and Checklist

    j.   Avineon Software Development Plan

    k.   Avineon Configuration Management Plan

    l.   Avineon Project Plan

    m. Avineon Quality Assurance Procedure

    n.   Avineon Quality Assurance Plan Template

    o.   Avineon Verification and Validation Procedures

    p.   Avineon Process Definition Procedure

    q.   Avineon Peer Review Procedures

1.5   Acronyms and Abbreviations

ACRONYM/ABBREVIATION	DEFINITION
ACWP	Actual Cost of Work Performed
BCWS	Budgeted Cost of Work Scheduled
BCWP	Budgeted Cost of Work Performed
BRD	Business Requirements Document
CCB	Change Control Board
CDR	Critical Design Review
CI	Configuration Item
CIO	Chief Information Officer
CLIN	Contract Line Item Number
CM	Configuration Management
CMP	Configuration Management Plan
CMM	Capability Maturity Model
CMMI	Capability Maturity Model Integrated
CO	Contracting Office
COR	Contracting Office Representative
COTS	Commercial Off The Shelf
CPI	Cost Performance Index
CR	Change Request
CSCI	Computer Software Configuration Item
CV	Cost Variance
DB	Database
DCF	Data Collection Form
DCG	Development Coordination Group
DMM	Deliverable Management Module
DOA	Date of Award
DR	Design Review
EAC	Estimate At Completion
ETC	Estimate To Complete
EV	Earned Value
EVA	Earned Value Analysis

EVM	Earned Value Management
FASS-PH	Financial Assessment Subsystem Public Housing
FCA	Functional Configuration Audit
FDS	Functional Design Specification
FEDSIM	Federal Systems Integration and Management Center
FOIA	Freedom Of Information Act
FQR	Formal Qualification Review
GAAP	Generally Accepted Accounting Principles
GAO	Government Accounting Office
GASB	Governmental Accounting Standards Board
GSA	General Services Administration
GTM	Government Technical Monitor
HTML	Hypertext Markup Language
HUDCAPS	HUD Central Accounting Processing System
HUD OIG	HUD Office of Inspector General
HUDWeb	HUD's Intranet Web Site
ICD	Interface Control Deliverable
IG	Inspector General
IPA	Independent Public Accountant
IPR	In Progress Reviews
ISG	Internet Services Group
IT	Information Technology
IV&V	Independent Verification & Validation
JAD	Joint Application Development
LOCCS	Line of Credit Control System
LPF	Late Presumptive Failure
MF	Multi-Family
NASS	Integrated Assessment Subsystem
NDS	Non-Developmental Software
ODC	Other Direct Costs
OMB	Office of Management and Budget
PASS	Physical Assessment Subsystem
PCA	Physical Configuration Audit

PDR	Preliminary Design Review
PH	Public Housing
PHA	Public Housing Agency/Public Housing Authority
PHAS	Public Housing Assessment System
PIH	Public and Indian Housing
PM	Project Manager
PMC	Project Monitoring and Control
PMP	Project Management Plan
PNR	Problem Notification Report
PP	Project Plan
PP&O	Project Planning & Oversight
PPQA	Product &Process Quality Assurance
PR	Problem Reports
PRR	Product Readiness Review
QA	Quality Assurance
QAG	Quality Assurance Guidelines
QAP	Quality Assurance Plan
QASS	Quality Assurance Subsystem
RAF	Risk Analysis Form
RASS	Residential Assessment Subsystem
REAC	Real Estate Assessment Center
REACS	Real Estate Assessment Center System
RM	Risk Management
RR	Requirements Review
SAC	PHAS invalidation action code
SCI	Software Configuration Item
SCR	Software Change Request
SDD	Software Design Description
SDF	Software Development File
SDL	Software Development Library
SDM	Software Development Methodology
SDP	System Decision Paper
SDR	Software Design Review

SMP	Software Measurement Plan
SOW	Statement Of Work
SPI	Schedule Performance Index
SQA	Software Quality Assurance
SR	Specification Review
SRS	Software Requirements Specification
SSDD	System/Subsystem Specification
SSR	Software Specification Review
SSS	System/Subsystem Specification
SV	Schedule Variance
SW	Software
TAC	Technical Assistance Center (formerly the Customer Service Center)
TBD	To Be Defined
TOR	Task Order Request
TOS	Tracking & Ordering System
TRB	Technical Review Board
TRR	Test Readiness Review
UAT	User Acceptance Testing
UDF	Unit Development Folder
UFI	Unique Fee Accountant Identifier
UII	Unique IPA Identifier
WASS	Web Access Security System
WBS	Work Breakdown Structure
WDDX	Web Dynamic Exchange
XML	eXtensible Mark-up Language

1.6   Points of Contact

1.6.1 Information

The following team members are listed as the points of contact on the FASS-PH program:

Type of contact	Contact name	Department	Telephone	e-mail
Government Technical Manager	Freddie Harrison	HUD PIH REAC	(202) 475-8639	Freddie_Harrison@hud.gov

Government Project Manager	Connie Ciccolella	GSA FEDSIM	(703) 306-7688	Connie.Ciccolella@gsa.gov
Government Contracting Officer	Shelly Goergen	GSA FEDSIM	(703) 306-7802	Shelly.Goergen@gsa.gov
Avineon Contract Manager	Barbara Neal	Avineon	(703) 671-1900	Bneal@avineon.gov
Avineon Senior Project Manager	Keith Bennett	Avineon	(703) 671-1900	Keith_Bennett@hud.gov

1.6.2 Coordination

Organization	Coordination Date	Support Function
PIH-REAC FASS-PH	Continuous, weekly meetings at minimum	Business requirements support, project management
PIH-REAC FASS-PH IT	Ongoing	Business/functional/system requirements, design, development, testing, maintenance
PIH-REAC DCG Development Coordination Group (DCG)	Continuous, weekly meetings at minimum	Development coordination, installation, deployment.
HUD IT	Ongoing	Implementation coordination, installation, deployment

1.7 Organization

Good software practice requires a measure of independence for the PPQA group. This independence provides a key strength to PPQA; that is, PPQA has the freedom, if the quality of the product is being jeopardized, to report, this possibility directly above the level of the project. While in practice this rarely occurs, for almost all problems are correctly addressed at the project level, the fact that the PPQA group can go above the project level gives it the ability to keep many of these problems at the project level. Figure 2 shows an example of a PPQA organization with relation to the project organization.
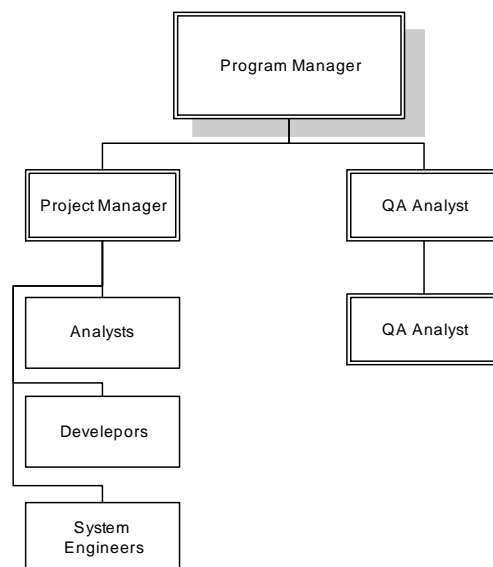
Figure 1 Reporting Structure Organization

The above organization diagram shows a sample reporting structure of two reporting lines for resolving PPQA issues. Although the QA analysts report directly to the Program Manager, rather than the Project Manager, they will interact directly with the HUD COTR/GTM and Project Manager. The Project Manager and/or HUD COTR/GTM have the authority to delegate responsibilities of interacting functions. The Project Manager reviews the PPQA Plan developed by the QA analyst with the final approval provided by the HUD COTR/GTM.

PPQA, comprising of QA analysts, is responsible for identifying compliance areas as either conforming or non-conforming with the PPQA policy and guidelines set forth in this plan with the goal of ensuring compliance with PPQA requirements. The PPQA organization assures the quality of deliverable software, and the engineering processes used to produce its documentation and non-deliverable software.

The following describes the functional groups that influence and control software quality.

    a. <u>Program Manager</u> is responsible for the following items:

        1. Establishing a quality program by committing the project to implement quality standards and methodologies based on CMM and HUD methodologies.

        2. Reviewing and approving the PPQA Plan applied to the FASS-PH subsystem development.

        3. Resolving and following-up on any quality issues raised by PPQA.

        4. Identifying an individual or group independent from the project to audit and report on the project's PPQA function.

        5. Identifying the quality factors to be implemented in the system and software.

    b. <u>Project Manager</u> is responsible for:

        1. Implementing the quality program in accordance with Avineon's PPQA policies, procedures, and guidelines as well as HUD's QA policies, procedures, and guidelines.

        2. Identifying the PPQA activities to be performed by PPQA.

3. Reviewing and approving the FASS-PH PPQA Plan applied to the FASS-PH subsystem development.

4. Assisting the Program Manager in the identification and funding of an individual or group independent from the project to perform the PPQA functions.

5. Resolving and following-up on any quality issues raised by PPQA.

6. Identifying and ensuring the quality factors to be implemented in the system and software.

7. Identifying, developing and maintaining planning documents such as the Project Management Plan, Test Plans, and this PPQA Plan.

c. <u>Project Team (Requirements Analysts)</u> is responsible for:

1. Reviewing and commenting on the FASS-PH PPQA Plan.

2. Implementing the quality program in accordance with this PPQA Plan.

3. Resolving and following-up on any quality issues raised by PPQA related to software development activities.

4. Identifying, implementing, and evaluating the quality factors to be implemented in the system (software and hardware).

5. Implementing engineering practices, processes, and procedures as defined in Avineon's CMM-based standard Technical Solution Procedure, Requirement's Development Procedures, Requirement's Management Procedures and HUD's Software Development Methodology.

d. Project Team (Software Developers) is responsible for:

1. Reviewing and commenting on the FASS-PH PPQA Plan.

2. Implementing the quality program in accordance with this PPQA Plan.

3. Resolving and following –up on any quality issues raised by PPQA related to software design and development.

4. Identifying, implementing, and evaluating the quality factors to be implemented in the software.

5. Implementing the software design/development practices, processes, and procedures as defined in Avineon's Software Development Procedures and HUD's Software Development Methodology.

e. Project Team (Software Test Engineers) is responsible for:

1. Reviewing and commenting on the FASS-PH PPQA Plan.

2. Implementing the quality program in accordance with this PPQA Plan.

3. Resolving and following-up on any quality issues raised by PPQA related to software test.

4. Verifying the quality factors are implemented in the system, specifically software.

5. Implementing the software test practices, processes, and procedures as defined in Avineon's Integration and Test Procedures and HUD's Software Development Methodology.

f. Project Team (CM Analysts) is responsible for:

1. Reviewing and commenting on the FASS-PH PPQA Plan.

2. Implementing the quality program in accordance with this PPQA Plan.

3. Resolving and following-up on any quality issues raised by PPQA related to CM.

4. Ensuring the quality factors are implemented in the software related to CM.

5. Implementing the CM practices, processes, and procedures as defined in Avineon's Configuration Management Policies and Procedures and HUD's Software Development Methodology.

g. Project Team (IV&V Analysts) is responsible for:

1. Reviewing and commenting on the FASS-PH PPQA Plan.

2. Implementing the quality program in accordance with this PPQA Plan.

3. Resolving and following-up on any quality issues raised by PPQA.

4. Verifying the quality factors are implemented in the system (hardware and software).

5. Implementing the practices, processes, and procedures as defined in Avineon's Validation and Verification Procedures and HUD's Software Development Methodology.

1.8   Resources

1.8.1   Facilities and Equipment

PPQA will have the same access to the facilities and equipment as the project team. In addition, PPQA will have access to computer resources to perform PPQA functions such as process and product evaluations and audits.

1.8.2   Personnel

The PPQA Analyst assigned to the project team has demonstrated experience in quality assurance as it relates to software engineering and software development. The PPQA Analyst has a marked and explicit understanding of CMM and its implications to the FASS-PH project, FASS-PH project team, and how it directly relates and results in quality results for REAC and HUD overall.

The PPQA Analyst will be responsible for ensuring the quality policies and procedures described in the PPQA Plan are executed.   Additionally, the PPQA will be familiar with software quality, software development-related activities, and structured analysis, design, coding, and testing.

2.0   SCHEDULE OF TASKS AND RESPONSIBILITIES

2   SCHEDULE OF TASKS AND RESPONSIBILITIES

The scheduling of PPQA tasks is driven by the project plan. Therefore, a PPQA task is performed in relationship to what software development activities are taking place. One or more PPQA tasks can be performed concurrently when a task is completed. A task is considered completed when the required report e.g., PPQA Reports, Process Audits Reports, etc. are satisfactorily completed or action items have been closed. The following tasks, requiring coordination and cooperation with the project team, shall be performed by PPQA as defined in Avineon's PPQA Procedures and HUD's Quality Assurance Guidelines and Software Development Methodology[1].

While conducting PPQA activities, the Avineon audit checklist will be used as a guide for conducting evaluations. The results of these tasks will be documented and reviewed with appropriate team members and submitted to HUD for review. Any recommended changes or corrective action will be approved in accordance with the configuration management procedures, before taking action.

2.1   Review Software Products

Per the Task Order Request GSC-TFMG-05-31210 for the FASS-PH project, the PPQA Analyst will evaluate and ensure software products are compliant with standards and guidelines for the following considerations:

**Table 1 Review Software Product Tasks Per Task Order Request
GSC-TFMG-05-31210 (Section C4-Statement of Work)**

Task #	PPQA Task
1.	Evaluate and ensure that FASS-PH subsystem's response time meets requirements stated in the Task Order Request (Section C.4.0.1)
2.	Evaluate and ensure that FASS-PH subsystem accessibility complies with Section 508 requirements stated in the Task Order Request (Section C.4.0.2).
3.	Evaluate and ensure that FASS-PH project planning complies with Project Planning requirements stated in the Task Order Request (Section C.4.1.1).
4.	Evaluate and ensure that the FASS-PH Transition Plan complies with Transition Plan requirements stated in the Task Order Request (Section C.4.1.2)
5.	Evaluate and ensure that the FASS-PH Risk Management Plan complies with Risk Management Plan requirements stated in the Task Order Request (Section C.4.1.3)
6.	Evaluate and ensure that the FASS-PH Quality Control Plan complies with Quality Control Plan requirements stated in the Task Order Request (Section C.4.1.4)
7.	Evaluate and ensure that the FASS-PH Progress Reports comply with Progress Reports requirements stated in the Task Order Request (Section C.4.1.5.1)
8.	Evaluate and ensure that FASS-PH Earned Value Analysis reporting shall include: Actual Cost of Work Performed (ACWP); Budgeted Cost of Work Performed (BCWP); Budgeted Cost of Work Scheduled (BCWS); Cost Performance Index (CPI); Cost Variance (CV); Earned Value (EV); Estimate At Completion (EAC); Estimate to Complete (ETC); Schedule Performance Index (SPI); Schedule Variance (SV) and complies with Earned Value requirements as stated in

---

[1]   A waiver exempting the project from any of these tasks is required.

	the Task Order Request (Section C.4.1.5.1.)
9.	Evaluate and ensure that FASS-PH Deliverables are entered into HUD's Deliverable Management Module (DMM) and complies with Deliverable Management Module requirements as stated in the Task Order Request (Section C.4.1.5.2)
10.	Evaluate and ensure that the FASS-PH project team conducts In Progress Review (IPR) Meetings and complies with In Progress Review requirements as stated in the Task Order Request (Section C.4.1.5.3)
11.	Evaluate and ensure that Problem Notification Reports are provided for any problem which may place in jeopardy any project component presented in the Project Management Plan and complies with Problem Notification Report requirements as stated in the Task Order Request (Section C.4.1.5.4)
12.	Evaluate and ensure that documentation complies with HUD Software Development Methodology and Document standards as stated in the Documentation requirements of the Task Order Request (Section C.4.1.5.5)
13.	Evaluate and ensure that all FASS-PH Reports comply with the Reporting requirements as stated in the Task Order Request (Section C.4.1.5.6)
14.	Evaluate and ensure that all FASS-PH Meetings comply with the Meeting requirements as stated in the Task Order Request (Section C.4.1.5.7)
15.	Evaluate and ensure that FASS-PH Planning activities and schedule comply with the Planning requirements as stated in the Task Order Request (Section C.4.2)
16.	Evaluate and ensure that FASS-PH Release 8.1 objectives comply with the Release 8.1 objectives as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
17.	Evaluate and ensure that FASS-PH Release 8.2 objectives comply with the Release 8.1 objectives as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
18.	Evaluate and ensure that FASS-PH Release 8.3 objectives comply with the Release 8.1 objectives as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
19.	Evaluate and ensure that FASS-PH Release 8.4 objectives comply with the Release 8.1 objectives as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
20.	Evaluate and ensure that FASS-PH Requirements Reviews comply with the Requirements Review requirements as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
21.	Evaluate and ensure that FASS-PH Design Reviews comply with the Design Review requirements as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
22.	Evaluate and ensure that FASS-PH Specification Reviews comply with the Specification Review requirements as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)

23.	Evaluate and ensure that FASS-PH Preliminary Design Reviews comply with the Preliminary Design Review requirements as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
24.	Evaluate and ensure that FASS-PH Critical Design Reviews comply with the Critical Design Review requirements as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
25.	Evaluate and ensure that FASS-PH Test Readiness Reviews comply with the Test Readiness Review requirements as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
26.	Evaluate and ensure that FASS-PH Formal Qualification Reviews comply with the Formal Qualification Review requirements as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
27.	Evaluate and ensure that FASS-PH Production Readiness Reviews comply with the Production Readiness Review requirements as stated in the FASS-PH Development Requirements section of the Task Order Request (Section 4.3)
28.	Evaluate and ensure that FASS-PH Release 8.1 Development Requirements comply with the Release 8.1 Development Requirements as stated in the Task Order Request (Section C.4.3.1)
29.	Evaluate and ensure that FASS-PH Release 8.2 Development Requirements comply with the Release 8.2 Development Requirements as stated in the Task Order Request (Section C.4.3.2)
30.	Evaluate and ensure that FASS-PH Release 8.3 Development Requirements comply with the Release 8.3 Development Requirements as stated in the Task Order Request (Section C.4.3.3)
31.	Evaluate and ensure that FASS-PH Operational Support complies with the Operational Support requirements as stated in the Task Order Request (Section C.4.4)
32.	Evaluate and ensure that FASS-PH Corrective Maintenance complies with the Corrective Maintenance requirements as stated in the Task Order Request (Section C.4.4)
33.	Evaluate and ensure that FASS-PH Adaptive Maintenance complies with the Adaptive Maintenance requirements as stated in the Task Order Request (Section C.4.4)
34.	Evaluate and ensure that FASS-PH Maintenance Requirements during the Base Period comply with Maintenance Requirements for the Base Period as stated in the Task Order Request (Section C.4.4.1)
35.	Evaluate and ensure that FASS-PH Maintenance Requirements during the Option Periods 1 and 2 comply with the Maintenance Requirements for Option Periods 1 and 2 as stated in the Task Order Request (Section C.4.4.2)
36.	Evaluate and ensure that FASS-PH Change Control Management activities comply with Change Control Management requirements as stated in the Task Order Request (Section C.4.5)

2.2   Evaluate Software Tools

PPQA shall conduct evaluations of tools, both existing and planned, used for software development and support. The tools are evaluated for adequacy by assessing whether they perform the functions for which the tools are intended and for applicability by assessing whether the tool capabilities are needed for the software development or support. Planned tools are evaluated for feasibility by assessing whether they can be developed with the techniques and computer resources available or by procurement.

2.3 Evaluate Facilities

PPQA shall evaluate facilities, both existing and planned, for adequacy by assessing whether they provide the needed equipment and space used for software development and support.[2]

2.4 Evaluate Software Products Review Process

This PPQA task assures that quality review processes are in place for all software products, which ma include representations of information other than traditional hard-copy documents, and that these products have undergone software product evaluation, testing, and corrective action as required by the standard.

PPQA shall check that software products that are ready for review are reviewed, results are verified and reported; issues or problems reported are resolved in accordance with Task Order Request GSC-TFMG-05-31210.

The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.5 Task: Evaluate Project Planning, Tracking and Oversight

Project planning, tracking and oversight involve project management to develop and document plans for Software Development, CI and System Test, Software Installation, and Software Transition. For project documents to be developed, PPQA will assist in identifying the appropriate guidelines, standards, or Data Item Description (DIDs), and will assist with the tailoring of those guidelines, standards, or DIDs to meet the project's needs.

PPQA will evaluate that the project conducts the relevant activities stated in the FASS-PH Project Management Plan. To verify that these activities are performed as planned, PPQA will audit the processes that define the activity, and will use any planning document as the measure of whether those activities are being met.

2.6 Evaluate Requirements Analysis Process

Requirements analysis establishes a common understanding of the customer's requirements between that customer and the software project team. An agreement with the customer on the requirements for the software project is established and maintained. This agreement is known as allocating system requirements to software and hardware.   Section 4 lists the system requirements documents.

PPQA activities involved in this task are listed below:

a. Verify that the correct participants are involved in the system requirements analysis process to identify all user needs.

b. Verify that requirements are reviewed to determine if they are feasible to implement, clearly stated, and consistent.

c. Verify that changes to allocated requirements; work products and activities are identified, reviewed, and tracked to closure.

---

[2] Facility evaluation reports will be submitted to the Project Manager if any inadequacies are identified.

d. Verify that project personnel involved in the system requirements analysis process are trained in the necessary procedures and standards applicable to their area of responsibility to do the job correctly.

e. Verify that the commitments resulting from allocated requirements are negotiated and agreed upon by the affected groups.

f. Verify that commitments are documented, communicated, reviewed, and accepted.

g. Verify that allocated requirements identified as having potential problems are reviewed with the group responsible for analyzing system requirements and documents, and that necessary changes are made.

h. Verify that the prescribed processes for defining, documenting, and allocating requirements are followed and documented.

i. Confirm that a configuration management process is in place to control and manage the baseline.

j. Verify that requirements are documented, managed, controlled, and traced (preferably via a matrix).

k. Verify that the agreed upon requirements are addressed in the SDP.

PPQA may use the Avineon audit checklists as a guide for conducting the evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.7 Evaluate System Design Process

The purpose of the system design process is to develop decisions about the system's behavioral design and other decisions affecting the selection and design of system components. System architectural design is organizing a system into subsystems, organizing a subsystem into Hardware Configuration Items (HWCIs), CIs and manual operations, or other variations as appropriate. Section 4 lists the system design documents.

PPQA activities are listed below:

a. Verify that system design documents and the traceability matrix are prepared and kept current and consistent.

b. Verify that relevant documents are updated and based on approved requirements changes.

c. Verify that design walkthroughs (peer reviews) evaluate compliance of the design of requirements, identify defects in the design, and evaluation and report design alternatives.

d. Participate in a sampled set of design walkthroughs and verify all walkthroughs are conducted.

e. Identify defects, verify resolution for previous identified defects, and verify change control integrity.

f. Selectively review and audit the content of system design documents

g. Identify lack of compliance with standards and determine corrective actions.

h. Determine whether the requirements and accompanying design and tools conform to standards, and whether waivers are needed prior to continuing software development.

  i. Review demonstration prototypes for compliance with requirements and standards.

  j. Verify that the demonstration conforms to standards and procedures.

  k. Review the status of design milestones.

PPQA may use the audit checklist as a guide for conducting the evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.8 Evaluate Software Requirements Analysis Process

The purpose of software requirements analysis is to formulate, document and manage the software requirements baseline; respond to requests for clarification, correction or change; analyze impacts; revise the software requirements specification; and manage software requirements analysis and change process. Section 4 lists the software requirements documents.

PPQA activities are listed below:

  a. Verify that the software requirements analysis process and associated requirements reviews are conducted in accordance with the standards and procedures established by the project and as described in Avineon's SDP and HUD's SDM.

  b. Verify that action items resulting from reviews of the software requirements analysis are resolved in accordance with these standards and procedures.

PPQA may use the audit checklist as a guide for conducting the evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.9 Evaluate Software Design Process

Preliminary design activity determines the overall structure of the software to be built. Based on the requirements identified in the previous phase, the software is partitioned into modules, and the function(s) of each module and relationships among these modules are defined.

A goal of detailed design is to define logically how the software will satisfy the allocated requirements. The level of detail of this design must be such that someone other than the original designer can accomplish the coding of the computer program. Section 4 lists the software design documents.

PPQA activities are listed below:

  a. Verify that the software design process and associated design reviews are conducted in accordance with standards and procedures established by the project and as described in Avineon's SDP and HUD's SDM.

  b. Verify that action items resulting from reviews of the design are resolved in accordance with these standards and procedures.

  c. Evaluate the method used for tracking and documenting the development of a software unit to determine the method's utility as a management tool for assessing software unit development progress. Example criteria to be applied for the evaluation are the inclusion of schedule information, results of audits, and an indication of internal review and approval of its constituent parts.

d. Verify that the method, such as the Software Development File (SDF) or Unit Development folder (UDF), used for tracking and documenting the development of a software unit is implemented and is kept current.

PPQA may use the audit checklist as a guide for conducting evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.10 Evaluate Software Code and Unit Testing Process

Software implementation or coding is the point in the software development cycle at which the design is finally implemented. The process includes unit testing of the software code.

PPQA activities are listed below:

a. Verify that the coding process, associated code reviews, and software unit testing are conducted in conformance with the standards and procedures established by the project and as described in Avineon's SDP and HUD's SDM.

b. Verify that action items resulting from reviews of the code are resolved in accordance with these standards and procedures.

c. Verify that the SDF used for tracking and documenting the development of a software unit is implemented and is kept current.

PPQA may use the audit checklist as a guide for conducting the evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.11 Evaluate Integration and Testing, System Qualification Testing, and Acceptance Testing Processes

Software integration and test activities combine individually developed components together in the testing environment to verify that they work together to complete the software and system functionality. For joint hardware and software development, integration requires close synchronization of hardware and software to meet designated integration and test milestones.

In the integration and test phase of the development lifecycle, the testing focus shifts from an individual component correctness to the proper operation of interfaces between components, the flow of information through the system, and the satisfaction of system requirements.

PPQA activities are listed below:

a. Verify that software test activities are identified, test environments have been defined, and guidelines for testing have been designed. PPQA will verify the software integration process, software integration testing activities and all software performance testing activities are being performed in accordance with Avineon's Validation and Verification policies and procedures as well as HUD's Software Development Methodology.

b. Verify any transfer of control of code to personnel performing software integration testing or software performance testing is being accomplished in accordance with Avineon's Validation and Verification policies and procedures as well as HUD's Software Development Methodology.

c. Verify that as many of the software integration tests as necessary and all software performance tests are witnessed to verify that the approved test procedures are being followed, that accurate records of test results are being kept, that all discrepancies discovered during the tests are being properly reported, that test results are being analyzed, and the associated test reports are completed.

d. Verify that discrepancies discovered during software integration and performance tests are identified, analyzed, documented, and corrected; software unit tests, and software integration tests are re-executed as necessary to validate corrections made to the code; and the software unit's design, code, and test is updated based on the results of software integration testing, software performance testing, and corrective action process.

e. Verify the software performance tests produce results that will permit determination of performance parameters of the software.

f. Verify that the responsibility for testing and for reporting on results has been assigned to a specific organizational element.

g. Verify that procedures are established for monitoring informal testing.

h. Review the Software Test Plan and Software Test Descriptions for compliance with requirements and standards.

i. Monitor test activities, witness tests, and certify test results.

j. Verify that requirements have been established for the certification or calibration of all support software or hardware used during tests.

PPQA may use the audit checklists in Figures 8 and 9 as guides for conducting these evaluations.

The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.12 Evaluate End-item Deliverable (Release) Process

PPQA shall evaluate the activities in preparation for end-item delivery to verify that program or project requirements for functional and physical audits of the end-item product are being satisfied. In some cases, the PPQA should be allowed to prohibit delivery of certain items, such as documentation, code, or a system, if the project fails to meet contractual requirements or standards.

PPQA may use the audit checklist as a guide for conducting this evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

The need for control increases proportionally to the number of individuals that use the products of software development. As a result, different control procedures will be used depending on use.

- Preliminary software releases are identified with numeric revision identification and will be used early in the development process. These release identifiers will be used on software deliverables when sufficient functionality has been developed and can be used by others outside of the immediate software development team. A release numbering system will be proposed to HUD for approval before implementation.

- Baseline software releases are identified with alphabetic revision identification will be used later in the development process at specified milestones. Again, a release numbering system will be proposed to HUD for approval before implementation.

2.13 Evaluate the Corrective Action Process

The corrective action process describes steps for (1) problem identification and correction occurring during software development to verify early detection of actual or potential problems, (2) reporting of the problem to the proper authority, (3) analysis of the problem to propose corrective measures, (4) timely and complete corrective action, and (5) the recording and follow-up of each problem's status. Problems in this context include documentation errors, software errors, and noncompliance with standards and procedures.

PPQA activities are listed below:

a. Periodically review the corrective action process and their results against the CM Plan to assess the effectiveness of the corrective action process.

b. Perform periodic analysis of all reported problems to identify trends that may disclose generic problem areas. These analyses shall include the study of the causes, magnitude of impact, frequency of occurrence, and preventive measures.

2.14 Evaluate Media Certification Process

PPQA shall verify that CM certifies that the media containing the source code, and the media containing the object code which are delivered to HUD, correspond to one another. PPQA shall also verify that all products delivered using HUD's Deliverable Management Module (DMM) and that all final deliverables will be in a format consistent with HUD Standards as described in the Task Order Request (Section C – Statement of Work). PPQA shall verify also that the software version represented by this media matches that on which software performance testing was performed, or correctly represents an authorized update of the code, as applicable.

PPQA may use the audit checklist as a guide for conducting this evaluation. PPQA reports, together with the corrective action records, software test reports, and software product evaluation records can constitute the required evidence for certification. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.15 Non-Deliverable Software Certification

The project may use non-deliverable software in the development of deliverable software as long as the operation and support of the deliverable software after delivery to HUD do not depend on the non-deliverable software or provision is made to verify that HUD has or can obtain the same software. PPQA shall certify that the use of non-deliverable software meets the above criteria, that is, deliverable software is not dependent on non-deliverable software to execute, or verify that HUD can obtain the same software. PPQA shall verify that all non-deliverable software used on the project performs its intended functions.

PPQA may use the audit checklist as a guide for conducting this evaluation. PPQA reports, together with the corrective action records, software test reports, and software product evaluation records can constitute the required evidence for certification.

PPQA may use the audit checklist as a guide for conducting this evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.16 Evaluate Storage and Handling Process

PPQA shall verify that there is an established plan, methodology, or set of procedures for storage and handling of media. PPQA shall evaluate the storage of the software product and documentation to verify that storage areas for paper products or media are free from adverse environmental effects such as high humidity, magnetic forces, and dust.

PPQA may use the audit checklist as a guide for conducting this evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.17 Evaluate Subcontractor Control

PPQA shall be responsible for ensuring that the quality of all software products from subcontractors conforms to the contract requirements and that the subcontractor's CM plan and procedures are being followed.

PPQA may use the audit checklist as a guide for conducting this evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.18 Evaluate Deviations and Waivers

PPQA shall assist program or project management, with requests for deviations and waivers, if required, and verify that the deviation or waiver request is processed in accordance with the project's CM Plan and approved by HUD.

PPQA may use the audit checklist as a guide for conducting this evaluation.

2.19 Evaluate Configuration Management Process

Configuration Management (CM) is the discipline that applies technical and administrative direction and surveillance to (1) identify and document the functional and physical characteristics of CIs, (2) control the changes to CIs and their related documentation, (3) record and report information needed to manage CIs effectively, including the status of proposed changes and the implementation status of approved changes, and (4) audit the CIs to verify conformance to specifications, interface control documents, and other contract requirements.

PPQA activities are listed below:

a. Verify that configuration identification of documents, code, and computer data has established standards for titling, naming, and describing change status.

b. Verify that baseline management of changes to the developmental baseline (including documents, code and computer data) are identified, reviewed, implemented, and incorporated in accordance with established procedures.

c. Verify configuration control of changes to baseline documents and software are being managed in accordance with CM requirements as stated in the CM Plan.

d. Verify configuration status accounting reports are prepared at established times, are prepared in accordance with established procedures, and report the status of items that are significant with respect to the management of the configuration of the software product and documentation.

e. Verify that the personnel assigned to participate in the configuration audits comply with the CM Plan.

f. Verify for document control that only approved, up-to-date documentation is provided for use by project personnel, and that the document distribution process results in receipt of correct documentation.

g. Verify that the program support library is the single place of storage for the baseline version of all software. Verify that the identification of all software includes the software name and a unique version identifier. The evaluation shall also determine that control of access to software products is being properly exercised and that unauthorized changes to master files cannot occur.

PPQA may use the audit checklist as a guide for conducting this evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.20 Evaluate Software Development Library Control Process

The Software Development Library (SDL) functions as the main control point for CM. A SDL contains all units of code developed for evolving project CIs, as well as carefully identified listings, patches, errata, CI and system magnetic tapes and disk packs, and job control streams for operating or building software systems. The SDL also contains previous version of the operational software system in the form of magnetic tapes or disk packs.

PPQA activities are listed below:

a. Verify the establishment of the SDL and procedures to govern its operation

b. Verify that documentation and computer program materials are approved and placed under library control.

c. Verify the establishment of formal release procedures for CM approved documentation and software versions.

d. Verify that library controls prevent unauthorized changes to the controlled software and verify the incorporation of all approved changes.

PPQA may use the audit checklist as a guide for conducting this evaluation. The results of this task shall be documented using the Process Audit Form described in Section 7 and provided to project management. PPQA recommendation for corrective action requires project management's disposition and will be processed in accordance with the guidance in Section 7.

2.21 Evaluate Non-Developmental Software

Non-Developmental Software (NDS) is software that is provided by the contractor, the Government, or a third party. NDS may be referred to as reusable software, Government-furnished software, or commercially available software depending on it source. PPQA shall verify that non-developmental software performs its intended functions.

PPQA may use the audit checklist as a guide for conducting this evaluation. PPQA reports, together with the corrective action records, software test reports, and software product evaluation records can constitute the required evidence for certifying the software performs its intended functions.

2.22 Verify Software Quality Assurance

The Project Manager requests periodic independent assessments of project PPQA. These assessments will be conducted on an annual basis or other agreed upon interval. The auditor, who is independent of the assessed PPQA group, will review PPQA audits conducted on the project, including documented findings and corrective actions, and will consult with project personnel to ensure that PPQA activities have been accomplished, and that corrective actions have been implemented or resolved. The auditor will report findings of the independent assessment to the Project and, where appropriate, Program Manager. Independent assessments may be requested of higher-level PPQA personnel (where available, Org-Level PPQA personnel) or from SEPG.

2.23 Risk Management

The FASS-PH has developed a risk management plan as identified in the FASS-PH Risk Management Plan. PPQA will review and evaluate the technical risk analysis and any risk reduction plan. PPQA reporting will confirm that the identified risks are managed in accordance with the provisions of the project's risk management plans, and that associated action items are reported, managed, and followed through to closure.

2.24 Responsibilities

The overall responsibility for the quality of the deliverables on the FASS-PH project resides with the Project Manager. The PPQA analyst will assist the Project Manager in ensuring the PPQA procedures are executed in accordance with the PPQA Plan.

PPQA derives its authority from the Project Manager through the HUD COTR/GTM. PPQA shall monitor project staff activities and review products for compliance to applicable standards, procedures, utilizing both HUD and Avineon's CMM based methodologies. The results of PPQA monitoring and analysis along with PPQA recommendations for corrective action shall be reported to the Project Manager, and, as required, to the HUD COTR/GTM. All documents and software approved by the Project Manager for release to HUD will have been reviewed and approved by PPQA. Table 4 is a responsibility matrix for the tasks identified in this Section.

Table 2 Responsibility Matrixes

PPQA Plan	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Develop/Document	X		X						

PPQA Plan									
Review PPQA Plan	X	X	X	X	X	X	X	X	X
Approve PPQA Plan	X	X	X						

Review Software Products	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Review products	X	X	X	X	X	X	X	X	X
Rework by author	Applies as applicable								
Approve product		X	X						

Evaluate Software Tools	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Evaluate tool	X			X		X	X		
Resolve Audit Findings		X	X						

Evaluate Software Facilities	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Evaluate facilities	X					X	X		
Resolve Audit Findings		X	X						

Proj Planning, Tracking & Oversight (PPT&O) Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Develop/Document SDP and other project plans (Test Plan, Training Plan, Computer Resource Life Cycle Management Plan (CRLCMP))		X	X						
Review plans	X	X	X	X	X	X	X	X	X
Approve plans		X	X						
Evaluate PPT&O Process	X								
Resolve Audit Findings		X	X						

System Requirements Analysis Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Develop/document Sys Rqmts					X				X
CM Sys Rqmts				X					
Review Sys Rqmts	X	X	X		X	X	X	X	X
Approve Sys Rqmts		X	X						
Evaluate/report Sys Rqmts Analysis Process	X								
Resolve Audit Findings		X	X						

System Design Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Develop/document Sys Design					X				
CM Sys Design				X					
Review Sys Design	X	X	X		X	X	X	X	X
Approve Sys Design		X	X						
Evaluate/report Sys Design Process	X								
Resolve Audit Findings		X	X						

Software Requirements Analysis Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Develop/document SW Rqmts						X	X		
CM SW Rqmts				X					
Review SW Design	X	X	X		X	X	X	X	X
Approve SW Rqmts		X	X						
Maintain SDL and SDFs				X	X	X			
Evaluate/report SW Rqmts Analysis Process	X								
Resolve Audit Findings		X	X						

Software Design Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Develop/document SW Design						X	X		
CM SW Design				X					
Review SW Design	X	X	X		X	X	X	X	X
Approve SW Design		X	X						
Maintain SDL and SDFs				X		X			
Evaluate/report SW Design Process	X								
Resolve Audit Findings		X	X						

Software Implementation & Unit Testing Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Develop/fix code						X			
CM code				X					
Code review	X					X	X		
Unit Test						X	X		
Maintain SDL and SDFs				X		X	X		
Maintain STR process				X					
Evaluate/report SW Implementation and Unit Testing Process	X								
Resolve Audit Findings		X	X						

Unit Integration and Testing Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Integrate SW						X			
Test Integrated SW							X	X	
Fix errors						X			
Maintain SDL and SDFs				X		X	X		
Maintain STR process				X					
Evaluate/report Unit Integration and Testing Process	X								
Resolve Audit Findings		X	X						

CI Qualification Testing Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Performance Test							X	X	
Fix errors						X			
Maintain SDL and SDFs				X		X	X	X	
Maintain STR process				X					
Evaluate/report CI Qualification Testing Process	X								
Resolve Audit Findings		X	X						

End-item Delivery Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Prepare/document version release doc				X					
Review version release doc	X				X	X	X	X	X
Approve version release doc			X						
Evaluate/report End-item Delivery Process	X								
Resolve Audit Findings		X	X						

Corrective Action Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Follow Corrective Action Process	X	X	X	X	X	X	X	X	X
Maintain Corrective Action Process				X					
Evaluate/report Corrective Action Process	X								
Resolve Audit Findings		X	X						
Certification (media certif., SW)	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Follow Certification Process	X			X			X	X	X
Certify SW	X			X					
Evaluate/report Certification Process	X								
Resolve Audit Findings		X	X						

Storage & Handling Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Follow Storage and Handling Process	X			X		X	X	X	X
Evaluate/report Storage and Handling Process	X								
Resolve Audit Findings		X	X						

Subcontractor Control	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Evaluate subcontractor software products	X		X	X	X	X	X	X	X
Evaluate/report Subcontractor Control Process	X								
Resolve Audit Findings		X	X						

Deviations & Waivers	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Document deviations & waivers		X	X						
Recommend Approval			X						
Approve		Major	Minor						
Evaluate/report Deviation & Waiver Process	X								
Resolve Audit Findings		X	X						

Configuration Management Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Develop/Document CM Plan				X					
Review CM Plan	X	X	X		X	X	X	X	X
Approved CM Plan		X	X	X					
Follow CM processes	X	X	X	X	X	X	X	X	X
Document CM procedures				X					
Evaluate/report CM Process	X								
Resolve Audit Findings		X	X						

SW Development Library Control Process	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Establish SDL				X					
Follow SDL procedures	X		X	X	X	X	X	X	X
Evaluate/report SDL Process	X								
Resolve Audit Findings		X	X						

Evaluate Non-Developmental SW	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Evaluate non-development SW	X				X	X	X	X	X
Evaluate/report Non-developmental SW Process	X								
Resolve Audit Findings		X	X						
Integrate non-development SW					X	X	X	X	X

Evaluate Non-Developmental SW	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Resolve integration errors					X	X	X	X	X

Configuration Audits	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Assist/perform configuration audits	X			X	X	X	X	X	X
Evaluate/report Configuration Audit Process	X								
Resolve Audit Findings		X	X						

Software Quality Assurance	PPQA Mgr	Prog Mgr	Proj Mgr	CM	Sys Eng	SW Dev	SW Test	Sys Test	Logistics
Appoint an independent PPQA Auditor		X							
Assist PPQA audits	X			X	X	X	X	X	X
Evaluate/report PPQA Audit Process	X								
Resolve Audit Findings	X	X	X						

2.25 Schedule

PPQA schedules are closely coordinated with the software development. Process audits will be performed at the beginning of each new phase of development to verify that the appropriate processes are correctly implemented as defined in the planning documents. In addition, spot-checks (unscheduled audits) will be made during each phase of development to verify that the processes and desktop procedures are being followed. At the completion of a software development phase, PPQA will review and report whether all steps required to transition to the next phase have been accomplished.

3.0   SYSTEM DOCUMENTATION

3   SYSTEM DOCUMENTATION

During the PPQA activities, all documentation generated throughout the lifecycle will be reviewed. The identified documentation listed will be that which control the development, verification and validation, use and maintenance of the software.

The PPQA Plan includes the following deliverables for review. The criteria that will be applied during review and evaluation of all lifecycle documents are those defined in the HUD SDM.

3.1   Documents by Phase

Table 5 lists the documents that will be produced and delivered by phase during the FASS PH project software development lifecycle.

Table 3 - Documents by Phase

Phase	Documentation
Initial	- Needs Statement - Project Plan (including WBS) - Configuration Management Plan - Quality Assurance Plan - Feasibility Study - Cost/Benefit Analysis - Risk Analysis
Define	- System Support and Acquisition Plan (initial) - Functional Requirements Document - Data Requirements Document - System Security and Privacy Plan - Internal Audit Plan - Project Plan (updated)
Design	- System/Subsystem Specifications - Database Specifications - Program Specifications - System Support and Acquisition Plan (final) - Validation, Verification, and Testing Plan (initial) - Project Plan (updated)
Build	- Installation and Conversion Plan (initial) - Test Plan

	• User's Manual
	• Operations Manual
	• Maintenance Manual
	• Validation, Verification, and Testing Plan (final)
	• Project Plan (updated)
Evaluate	• Test Results and Evaluation Reports
	• Installation and Conversion Plan (final)
	• Project Plan (updated)
Operate	• Pilot Test Results
	• Project Plan (updated)

3.2   Discipline for Documentation Standard Practices

Documentation Standards shall adhere to Avineon's Documentation Standard Practices as well as HUD's Documentation Standards and Checklist. HUD's SDM Documentation Templates will be used as the basis for all documents deliverables of the FASS-PH project to HUD. Avineon's Documentation Standards will be utilized as an enhancement measure to documentation when appropriate. A waiver exempting project documentation deliverables from HUD's standard template usage will be required.

4.0   REVIEWS AND AUDITS

4   REVIEWS AND AUDITS

In order for PPQA to evaluate compliance with the PPQA Plan, PPQA will review and approve deliverables throughout the development process. These reviews will specify that the evidence of work generated is adequate to ensure compliance with project and contract requirements. Audits performed shall include examination of both internal and external software deliverables.

4.1   Verify Technical Reviews

A primary component of engineering quality into software is the conduct of technical reviews of software products, both deliverable and non-deliverable. Participants of a technical review shall include persons with technical knowledge of the software products to be reviewed. The purpose of the technical review will be to focus on in-progress and final software products rather than the materials generated especially for the review. PPQA will assure that technical reviews are accomplished and will selectively attend them in accordance with approved sampling techniques. The guidelines of Technical Reviews and Audits for Systems, Equipments, and Computer Software may be used for conducting a technical review.

A summary of each kind of technical review appears below:

a.   Requirements Review (RR): The RR is conducted during the Define System phase to ascertain the adequacy of the developer's efforts in defining requirements.

b.   Design Review (DR): The DR is conducted during the Define System phase to evaluate optimization, correlation, completeness and risks associated with allocated technical requirements, and to include a summary review of the system engineering process that produced the allocated technical requirements and of the engineering planning for the next phase of effort.

c.   Specification Review (SR): The SR is conducted during the Design System phase to review the finalized Configuration Item (CI) requirements and operational concept. A successful SR shows that there is a satisfactory basis for proceeding into preliminary design.

d.   Preliminary Design Review (PDR): The PDR is conducted during the Design System phase to evaluate progress, consistency and technical adequacy of the selected top-level design and test approach, to evaluate the compatibility between requirements and preliminary design, and to assess the preliminary version of the operation and support documents.

e.   Critical Design Review (CDR): The CDR is conducted during the Design System phase to determine acceptability of the detailed design, performance and test characteristics of the design solution, and the adequacy of operation and support documents.

f.   Test Readiness Review (TRR): The TRR is conducted during the Build System phase to determine whether the test procedures are complete and to assure that the developer is prepared for formal CI testing.

g.   Formal Qualification Review (FQR): The FQR is conducted during the Build System phase to determine that a group of configuration items comprising the system are verified to have met specific program or project management performance requirements through tests, inspections or analytical processes.

h.   Production Readiness Review (PRR): The PRR is conducted during the Evaluate System phase to determine the status of completion of the specific actions that must be satisfactorily accomplished prior to executing a production decision to go forward.

Table 4 Reviews and Audits[3]

SYSTEM AND SOFTWARE DEVELOPMENT PHASE	SOFTWARE PRODUCTS	REQUIRED AUDITS AND REVIEWS
Initiate Phase	1. Needs Statement 2. Feasibility Study 3. Risk Management Plan 4. Cost/Benefit Analysis 5. Systems Decision Paper 6. CM Plan 7. PPQA Plan	1. Management Review 2. Peer Review
Define Phase	1. Functional Requirements Document 2. Data Requirements Document 3. System Support Plan 4. System Security Plan 5. Internal Audit Plan	1. Requirements Review 2. Design Review 3. Process Audits 4. Management Review 5. Peer Review
Design Phase	1. System/Subsystem Specification 2. Database Specification 3. Program Specification 4. IV&V Plan	1. Specification Review 2. Preliminary Design Review 3. Critical Design Review 4. Process Audits 5. Management Review 6. Peer Review

---

[3]   Peer Reviews are discussed in Section 4.

SYSTEM AND SOFTWARE DEVELOPMENT PHASE	SOFTWARE PRODUCTS	REQUIRED AUDITS AND REVIEWS
Build Phase	1. Database 2. Computer Programs 3. User's Manual 4. Operations Manual 5. Maintenance Manual 6. IV&V Test Scripts 7. Test Plan (Unit and Integration) 8. Installation & Conversion Plan	1. Test Readiness Review 2. Formal Qualification Review 3. Functional Configuration Audit 4. Physical Configuration Audit 5. Process Audit 6. Management Review 7. Peer Review
Evaluate Phase	1. Test Results & Evaluation Reports 2. Installation and Conversion Plan (updated)	1. Production Readiness Review 2. Functional Configuration Audit 3. Physical Configuration Audit 4. Review Process Audits 5. Management Review 6. Peer Review
Operate Phase	1. User's Manual (updated) 2. Maintenance Manual (updated) 3. Production System	1. Physical Configuration Audit 2. Functional Configuration Audit 3. Process Audits 4. Management Review 5. Peer Review

Technical reviews will be conducted to review evolving software products, demonstrate proposed technical solutions, and provide insight and obtain feedback on the technical effort. The outcome of a technical review is listed below:

a.   Identify and resolve technical issues.

b.   Review project status, specifically surface near- and long-term risk regarding technical, costs, and schedule issues.

c.   Arrive at agreed-upon mitigation strategies for identified risks, within the authority of those present.

d.   Identify risks and issues to be raised at joint management reviews.

e.   Verify on-going communications between acquirer and developer technical personnel.

The entrance criteria for a technical review will require that an item to be reviewed be distributed to the group prior to the review meeting. Additionally, a recorder will be assigned to record any issues requiring resolution stating action item assignee and due date, and decisions made within the authority of the technical review participants.

Various measurements are collected as part of technical reviews to help determine the effectiveness of the review process itself as well as the process steps that are used to produce the item being reviewed. These measurements, reported to the project manager, will include the amount of time spent by each person involved in the review, including preparation for the review.

4.2   Verify Management Reviews

PPQA periodic management review of software project status, progress, problems, and risk will provide an independent assessment of project activities. PPQA will provide the following information to management:

   a.   Compliance - Identification of the level of compliance of the project with established organizational and project processes.

   b.   Problem areas - identification of potential or actual project problem areas based on analysis of technical review results.

   c.   Risks - identification of risks based on participation and evaluation of project progress and trouble areas.

Because the PPQA function is integral to the success of the project, PPQA will freely communicate its results to senior management, project management and the project team. The method for reporting compliance, problem areas, and risks will be communicated in a documented report or memorandum. Compliance, problem areas, and risks will be followed-up and tracked to closure.

4.3   Conduct Process Audits

Software development processes are audited according to the tasks specified in this Section and performed in accordance with the software development schedule specified in the SDP.

4.4   Conduct Configuration Audits

4.4.1 Functional Configuration Audit.

The Functional Configuration Audit (FCA) is held prior to software delivery to compare the software as built (including its executable forms and available documentation) with the software requirements as stated in the baseline Software Requirements Specification (SRS). The purpose is to ensure that the code addressed all, and only, the documented requirements and functional capabilities stated in the SRS. Avineon Configuration Management Procedure along with HUD's Quality Assurance Guidelines provides the direction for conducting an FCA. PPQA will participate as a member of the FCA team with other FCA team members to be assigned by the project manager. PPQA will assist in the preparation of the FCA findings. Any follow-up to the reported FCA finding will be monitored and tracked to closure.

4.4.2 Physical Configuration Audit.

The Physical Configuration Audit (PCA) is held to verify that the software and its documentation are internally consistent and are ready for delivery. The purpose is to assure that the documentation to be delivered is consistent and correctly describes the code. Avineon PPQA Procedure along with HUD's Quality Assurance Guidelines provides the direction for conducting a PCA. PPQA will participate as a member of the PCA team with other PCA team members to be assigned by the project manager. PPQA will assist in the preparation of the PCA findings. Any follow-up to the reported PCA finding will be monitored and tracked to closure.

4.5   Scheduled Audits

PPQA will generate and maintain an Audit Schedule. Audits will occur at the end of each development phase as indicated in the Project Plan. The results of audits will be discussed with the PM and the individual(s) responsible for the production of the deliverable. Results will be submitted to the HUD COTR/GTM in scheduled status reports.

4.6   Unscheduled Audits

PPQA will perform random and unannounced audits to ensure the corrective actions agreed to during the Scheduled Audits are being followed. The results of audits will be discussed with the PM and the individual(s) responsible for the production of the deliverable.

4.7   Audit Reports

Audit reports, and recommended corrective actions generated by PPQA will be brought to the attention of the individual(s) responsible for producing the software deliverable. Corrective action will be recommended and reviewed with the individual and PM. The results of audits of the PPQA function will be tracked and maintained by the PM.

5.0   TESTING

5 TESTING

FASS-PH testing activity includes unit level testing, integration testing (at Unit and CI/HWCI level), performance testing (CI Qualification Testing), and acceptance testing (System Qualification Testing). Figure 2 provides the Test Process Flow. PPQA shall audit the testing activities as defined in the Independent Validation & Verification Plan (IV&V Plan), and shall verify that the software and test documentation is subject to configuration management control. PPQA shall witness the tests and verify that test results are recorded and evaluated. PPQA shall coordinate the maintenance of Problem/Change Report (P/CR), sometimes called Software Trouble Report (STR), logs with CM and shall verify that software changes are controlled according to the Configuration Management and IV&V plans. PPQA shall witness regression-testing resulting from P/CRs or STRs to verify the effectiveness of the correction.
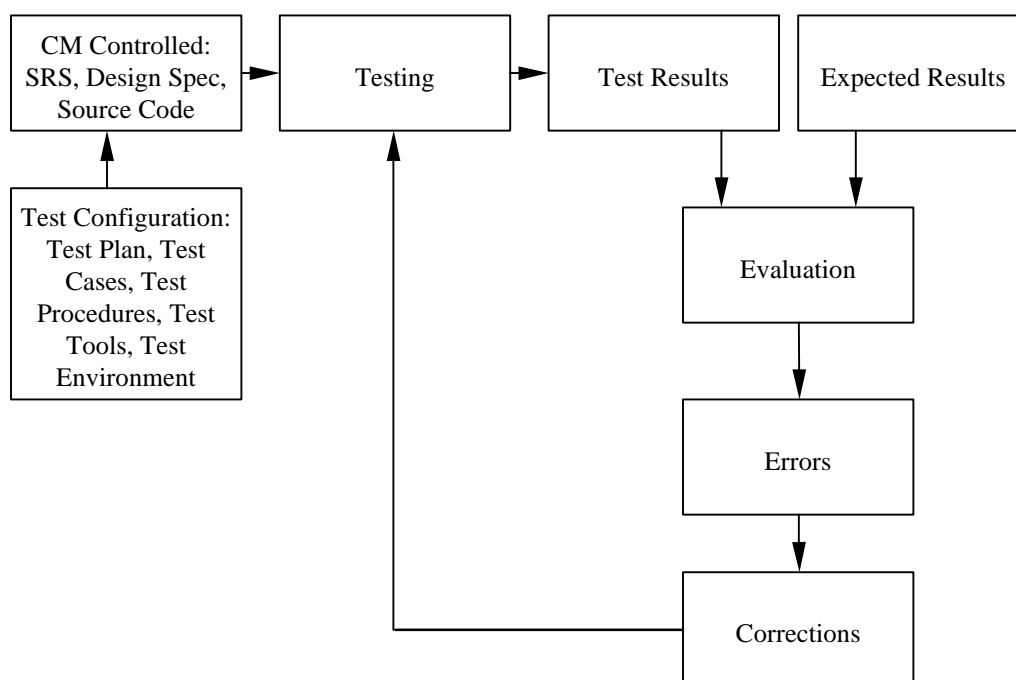
Figure 2 Testflow Process

6.0   PROBLEM REPORTING AND CORRECTIVE ACTION

6   PROBLEM REPORTING AND CORRECTIVE ACTION

This section describes the reporting and control system used by PPQA to record and analyze discrepancies and to monitor the implementation of corrective action. The forms utilized by PPQA for reporting include the Process Audit Report, P/CR or STR, Software Tool Evaluation Report, and Facilities Evaluation Report. Each form and their uses are discussed in the following section.

6.1   Process Audit Report

PPQA reports the results of a process audit and provides recommendations, if necessary, using the Process Audit Report. The Process Audit Report is used to record that the process is (1) being followed correctly and working effectively, (2) being followed but is not working effectively, (3) or not being followed.   Figure 3 provides the format of a Process Audit Report

6.1.1   Submittal and disposition of Process Audit Report

The Process Audit Report is directed to the groups listed below:

   a.   Senior Management - The results of process audits are used in conjunction with other project status information to guide senior management attention to identify and mitigate project risks at the organizational level.

   b.   SEPG - The SEPG utilizes the process audits results, in conjunction with the results of audits of other projects, to identify process weaknesses and initiate or enhance process improvement in specific areas. This data becomes part of the process database so that it is available for future project analysis and use.

   c.   Project Manager - The project manager utilizes the report in the ways listed below:

   1.   To provide insight into whether there is compliance with the development process and its effectiveness in meeting project goals. Where necessary and appropriate, the project manager may initiate enforcement activities or initiate change to the established processes using the approved procedures.

   2.   To indicate agreement, disagreement, or deferral of recommendations cited in the Process Audit Report. Should the Project Manager indicate disagreement with the recommendations recorded on the Process Audit Report, the final disposition of report recommendations is made by the appropriate Project Sponsor as described in Section 7.1.2.

PROCESS AUDIT REPORT

TRACKING IDENTIFIER:_____

LEAD AUDITOR:_____ DATE OF REPORT:_____

AUDIT TEAM:_____

_____

PROJECT NAME:_____

DATE OF AUDIT:_____

PROCESS/PROCEDURE AUDITED:_____

AUDIT CHECKLIST USED: (Attach)_____

AUDIT FINDINGS: (Check one.)

_____ Process/Procedure Acceptable

_____ Process/Procedure Conditionally Acceptable

(Subject to satisfactory completion of action items listed below)

Conditions noted:

_____ Process/Procedure Unacceptable

(Subject to satisfactory completion of action items listed below)

Conditions noted:

_____

__ACTION ITEM (AI):

AI #   TITLE   ASSIGNED TO: DUE DATE: COMP DATE:___

_____

_____

_____

CORRECTIVE ACTION:

_____

DISPOSITION: APPROVE   CANCEL   DEFER

Project Manager: DATE:

_____

AI CLOSURE:

PPQA Sign-off: DATE:

(FILE COMPLETED FORM IN PPQA EVALUATION RECORD.)

Figure 3 Process Audit Report

6.1.2 Escalation Procedure for resolution of Non-CONFORMANCE ON process Audit Report

In the event that affected project personnel dispute the findings and recommendations of a Process Audit Report, PPQA will first communicate with the Project Manager to resolve the dispute. If the Project Manager also disputes the findings and/or recommendations, the Program Manager directs final disposition of Process Audit Report recommendations. The affected project implements, defers, or cancels the implementation of corrective actions recommended on the Process Audit Report as directed by the Program Manager. This direction is recorded and dated by the Program Manager to be added to the PPQA evaluation records of the project.   PPQA retains the original record of findings and subsequent resolution data in its audit files.

6.2   Recording Problems in Software Code or Documentation

Issues found in the software code or documentation that is under configuration management must be recorded by means of a Software Trouble Report, regardless of how or by whom the problem was discovered. STRs generated by PPQA shall be prepared and processed in accordance with the documented procedure. PPQA shall analyze STRs for problem trends in an effort to prevent recurring discrepancies. PPQA will report the results of STR trend analyses along with suggestions for problem resolution and prevention.

6.3   Software Tool Evaluation Report

Figure 4 provides the format for evaluating software tools as described in Section 3.2.

SOFTWARE TOOL EVALUATION

PPQA:_____   DATE OF EVALUATION:_____

Software Tool Evaluated:

Methods or criteria used in the evaluation:

Evaluation Results:

Recommended Corrective Actions

Corrective Action Taken

Figure 4. Software Tool Evaluation

6.4   Facilities Evaluation Report

Figure 5 provides the format for evaluating existing and planned [[Project Name]] facilities as described in Section 3.3.

PROJECT FACILITIES EVALUATION

PPQA:_____DATE OF EVALUATION:_____

Facility Evaluated (Equipment, User/Test/Library Space):

Methods or criteria used in the evaluation:

Evaluation Results:

Recommended Corrective Actions

Corrective Action Taken

Figure 5. Project Facilities Evaluation

7.0   TOOLS

7   TOOLS

Many PPQA software tools are available to automate PPQA functions. Examples include, but are not limited to:

- Operating system utilities

- Debugging aids

- Documentation tools

- Code and structure analyzers

- Performance Monitors

- Statistical analysis packages

- Static or dynamic test tools

- Checklists,

- Standards auditors

- Simulators

- Execution analyzers

- Software traceability matrices

- Test drivers

- Test case generators

- Information engineering CASE tools.

Techniques include review of the use of standards, software inspections, requirements tracing, requirements and design verification, reliability measurements and assessments, and rigorous or formal logic analysis.

Methodologies are an integrated set of the above tools and techniques. The methodologies should be well documented for accomplishing the task or activity and provide a description of the process to be used.

8.0   PROJECT CONTROLS

8 PROJECT CONTROLS

8.1 Product Control

The purpose of this section is to define the methods and facilities used to maintain, store, secure and document controlled versions of the software artifacts during all phases of the software life cycle whose appropriate use will be verified by PPQA. This may be implemented in conjunction with an available CM tool, such as Rational ClearCase or PVCS. This activity will be in coordination with configuration management.

Code control includes the items listed below:

    a.   Identifying, labeling, and cataloging the software to be controlled

    b.   Identifying the physical location of the software under control

    c.   Identifying the location, maintenance, and use of backup copies

    d.   Distributing controlled copies of the code

    e.   Identifying the documentation that is affected by a change

    f.   Establishing a new version

    g.   Regulating user access to the code.

FASS-PH project will use PVCS for code control. The code control method is described in HUD's configuration management guidelines and software development methodology. PPQA will conduct ongoing evaluations of the code control process to verify that the process of controlling the code is effective and in compliance with Avineon's Configuration Management Policies and Procedures and HUD's Configuration Management Policies and Procedures. Section 3.19 further describes PPQA activities for verifying the CM process.

9.0   RECORD COLLECTION, MAINTENANCE, AND RETENTION

9 RECORDS COLLECTION, MAINTENANCE, AND RETENTION

The PPQA plan will be maintained and retained by the FASS-PH PPQA Analyst using PVCS Version Control. HUD facilities will be used to assemble, safeguard, and maintain this documentation, and designate the retention period.

Records and reports that provide a history of product quality throughout the software life cycle document PPQA activities. Measurement data collected will be reviewed for trends and process improvement. All PPQA records will be collected and maintained for the life of the project.

10.0 STANDARDS, PRACTICES, CONVENTIONS, METRICS

10   STANDARDS, PRACTICES, CONVENTIONS, METRICS

To verify the delivery of a fully conforming, high-quality product, every individual assigned to the project will participate in quality assurance. Avineon's methodologies define the procedures by which the software development staff verifies the quality of the product during the development process. This remainder of this section describes the procedures used by PPQA to verify that the quality assurance provisions of this PPQA Plan and applicable standards, practices, conventions, and metrics are met.

Avineon and HUD software development standards will be applied to the FASS-PH subsystem development project. If any standards require tailoring, they will be documented accordingly.

Section 3 identifies PPQA evaluation of the requirements, design, implementation, and test phase to verify compliance with Avineon's and HUD software development standards.

PPQA will verify the source code complies with the standards as detailed in Avineon's and HUD software development standards. Standards and practices for testing are described in Avineon Technical Solution Procedure. PPQA will verify testing activities are in compliance with Avineon Verification Procedure and HUD's Software Development Methodology.

10.1 Metrics

The following measurements will be made and used to determine the cost and schedule status of the PPQA activities:

a.   PPQA milestone dates (planned)

b.   PPQA milestone dates (completed)

c.   PPQA work scheduled (planned)

d.   PPQA work scheduled (actual)

e.   PPQA work completed (actual)

f.   PPQA effort expended (planned)

g.   PPQA effort expended (actual)

h.   PPQA funds expended (planned)

i.   PPQA funds expended (actual)

j.   Number of noncompliance items open

k.   Number of noncompliance items closed

l.   Total Number of noncompliance items

PPQA is responsible for reporting these measurements to the Project Manager on a monthly basis.

Measurements will be made by making a numerical assignment to specific attributes of each above-identified project component selected for measurement and/or any other project component that management believes is a candidate for measurement and reporting. The objective of measuring these components is to help management better understand the project and the relationship of project components to each other as well as take corrective action to ensure the success of the project if necessary.

The information content of a numerical assignment to a project component and its attributes is dependent on the scale of measurement used. The scale in the table below will determine the type of statistical analyses that will be employed in the study of the measurements taken. The scale of measurement is critical to validate the method of analysis for each measured project component.

The four measurement scales are:

- Nominal
- Ordinal
- Interval
- Ratio

SCALE	DEFINITION	EXAMPLE	STATISTICS
Nominal	Only the presence/absence of an attribute; can only count items	Go/no go; success/fail; accept/reject (i.e., critical defects per system test cycle)	Percent; proportion; chi-square tests
Ordinal	Can say that one item has more or less of an attribute than another item; can order a set of items	Risk inclination	Rank-order correlation
Interval	Difference between any two successive points is equal; often treated as a ratio scale even if assumption of equal intervals is incorrect; can add, subtract, order objects	Calendar time	Correlations; t-tests; multiple regression
Ratio	True zero point indicates absence of an attribute; can add, subtract, multiply and divide	Elapsed time; value-weight	t-test; correlations; multiple regressions

Nominal

Numbers on a nominal scale are not truly measurements. They are simply category labels in numerical forms (1 = Nonconformance; 2 = Conformance). Nominal scales represent the simplest form of measurement. They are most often viewed as a form of classification rather than measurement. The only mathematical operations permitted on nominal scales are "=" (shows the presence of an attribute) or "≠" (shows the absence of an attribute).

Ordinal

An ordinal variable is one that has a natural ordering of its possible values, but for which the distances between the values are undefined. An example is a customer requirement with a ranking of exceeds

customer expectations, meets customer expectations, does not meet customer expectations. Ordinal data can be analyzed with the mathematical operators, "=" (equality), "≠" (inequality), ">"(greater than), and "<" (less than). There are a wide variety of statistical techniques, which can be applied to ordinal data. Ordinal models include:

- Odds-ratio measures
- Log-linear models

Both of the above models are used to analyze cross-classifications of ordinal data presented in contingency tables. Ordinal data are commonly converted into nominal data and analyzed using binomial models. Using the example previously mentioned, this method would be employed to plot a *p* chart of items in each category (exceeds customer expectations, meets customer expectations, does not meet customer expectations).

Interval

Interval scales consist of measurements where the ratios of differences are invariant. Conversion between two interval scales is accomplished by the transformation $y = ax + b$, $a > 0$. As with ratio scales, when permissible transformations are made statistical, results are unaffected by the interval scale.

Ratio

Ratio scale measurements are so called because measurements of an object in two different metrics are related to one another by an invariant ratio. This implies that a change from one ratio measurement scale to another is performed by a transformation of the form $y = ax, a \neq 0$. When permissible transformations are used, statistical results based on the data are identical regardless of the ratio scale used.

Reliability and Validity

Any measurement taken of a project component must meet the following criteria:

- The measurement measures what it is intended to measure (i.e., it is valid).
- A re-measurement would order individual responses in the same way (i.e., it is reliable).
- The numerical assignment for a selected project component should be close to the actual property being measured (i.e., it is accurate).
- The measurement selected for a project component should be repeatable for that project component.

Process Audits are conducted throughout the lifecycle on pre-determined processes and work products. This audit will include measurement of these project components. The difference between the average measured value of the project components and the reference value (standard for project component) will be measured at the end of each release as part of lessons-learned.

- Processes and work products that will be measured include:
- Estimated LOE per requirement and Actual LOE per requirement
- Total Estimated LOE per release and Actual LOE per release
- Projected Risk and Realized Risk
- Number of Unknown Issues not identified in preliminary software risk evaluation sessions and sources compared to identified risks and sources
- Corrective Actions taken and number of Corrective Actions closed
- Number of critical, major, and minor software defects

- Number of changes made to requirements after RTM sign-off

11.0 TRAINING

11   TRAINING

All members of the PPQA function are experienced in quality assurance activities. To supplement their formal training and experience, they will be trained on HUD specific activities to ensure a smooth transition and overall effective PPQA program. As required, other project team members will be provided with training to support the PPQA activities. For example, the development group will be provided peer review training to conduct reviews on other developers' code. This will provide them information on what elements to look for in a peer review.

Any PPQA training requirements will be coordinated with the PM and PPQA. For example, training may be required for conducting code/peer reviews, documentation reviews, process audits, data and metrics collection, and management reviews.

Training may be conducted in several formats, including formal classroom training, brown bag lunch sessions, and On-the-Job (OJT) training.

Table 6 provides a matrix that identifies the required skills to perform PPQA tasks to implement the FASS-PH PPQA Plan. The training schedule will be compatible with the project schedule. In some cases, training will be conducted as On-the-Job (OJT) training.

Table 5 PPQA Required Skills and Tasks

TASK	SKILL REQUIREMENTS	TYPE	SOURCE
Code Reviews	Source Language, Peer Reviews	Classroom/ OJT	SEPG, Peer Review Process and Workshop
Documentation Reviews	Software Development and Documentation standards and guidelines, Peer Reviews	Classroom/ OJT	SEPG, Peer Review Process and Workshop
Process Audits	Software Development Life Cycle Processes, Audit techniques	Classroom/ OJT	Avineon SDP Procedure
Testing	Testing Methodologies	OJT	Avineon IV&V Procedure
PPQA Management	Project Management	Classroom/ OJT	SEPG, Software Project Management (PM) course
Metrics	Data Collection and Analysis	Classroom/ OJT	SEPG, PM course
Problem reporting and correction action	Configuration Management	Classroom/ OJT	SEPG, CM Practitioner's Training
Tools	Vendor supplied training	Classroom/ OJT	Vendor
Code, Media, and Supplier Control	Configuration Management	Classroom/ OJT	SEPG, CM Practitioner's Training
Risk Management	Risk Management Process	Classroom/	SEPG, PM course

and Analysis		OJT	
Software Management	Software Management Process	Classroom/ OJT	SEPG, Software Management for Everyone (SME) Training, PM course